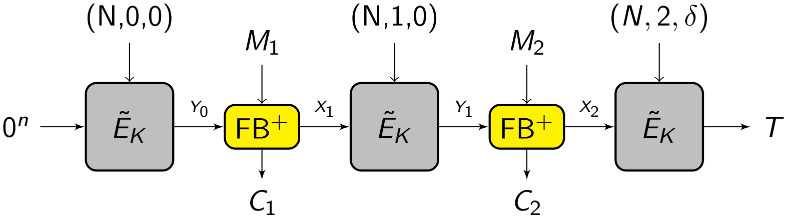
Security Analysis of mixFeed

Bishwajit Chakraborty and Mridul Nandi
Indian Statistical Institute,Kolkata

6th Nov 2019

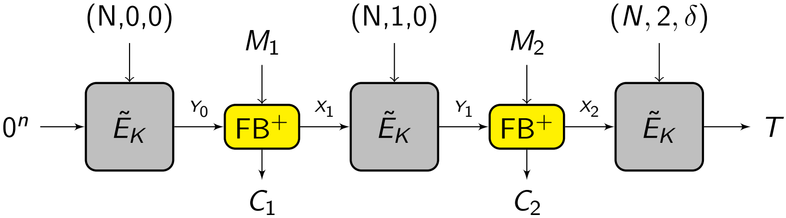
TBC-based AE Mode

- Simplified AE (with no AD) based on a TBC $\tilde{E}_K$.



TBC-based AE Mode

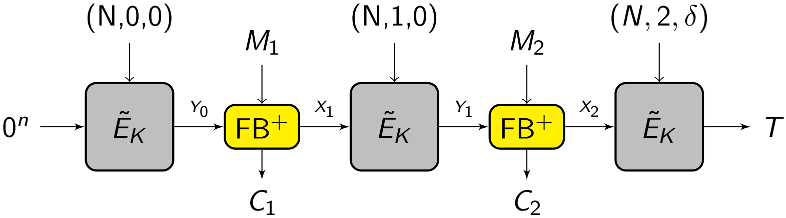
- Simplified AE (with no AD) based on a TBC $\tilde{E}_K$.



- State size: (i) TBC state n, (ii) Tweak and Key state $t + k$, (iii) Possibly additional state to hold t-bit tweak and k bit key.

TBC-based AE Mode

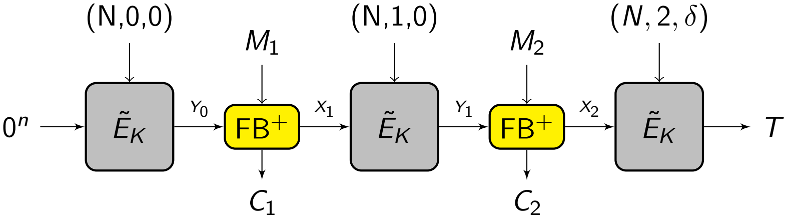
- Simplified AE (with no AD) based on a TBC $\tilde{E}_K$.



- State size: (i) TBC state n, (ii) Tweak and Key state $t + k$, (iii) Possibly additional state to hold t-bit tweak and k bit key.
- Decryption: FB^-(instead of FB^+).

TBC-based AE Mode

- Simplified AE (with no AD) based on a TBC $\tilde{E}_K$.

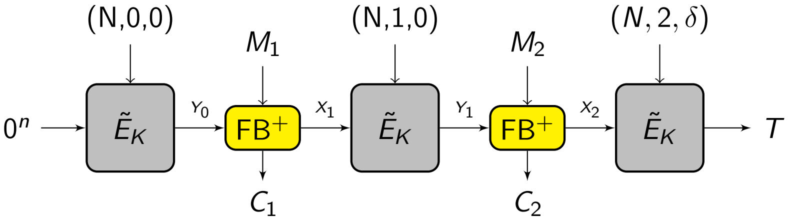


- State size: (i) TBC state n, (ii) Tweak and Key state $t + k$, (iii) Possibly additional state to hold t-bit tweak and k bit key.

- Decryption: FB^-(instead of FB^+).

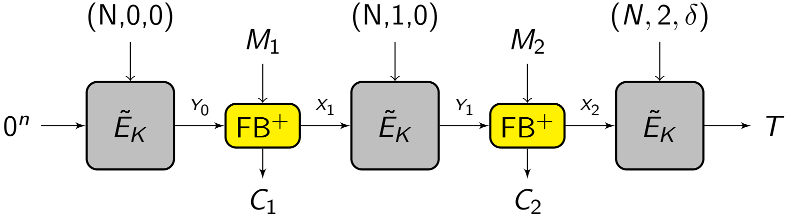
- Assume $C_i = M_i \oplus Y_{i-1}$ and X_i is dependent on Y_{i-1} and significant fraction of bits of C_i.

TBC-based AE Mode



$\mathbf{Adv}_{AE}^{priv}(D, T) \leq \mathbf{Adv}_{\tilde{E}}^{TPRP}(D, T)$ and $\mathbf{Adv}_{AE}^{auth}(D, T) \leq \mathbf{Adv}_{\tilde{E}}^{TPRP}(D, T) + \mathcal{O}(\frac{D}{2^n})$

TBC-based AE Mode



$$\mathbf{Adv}^{priv}_{AE}(D,T) \leq \mathbf{Adv}^{TPRP}_{\tilde{E}}(D,T) \text{ and } \mathbf{Adv}^{auth}_{AE}(D,T) \leq \mathbf{Adv}^{TPRP}_{\tilde{E}}(D,T) + \mathcal{O}(\tfrac{D}{2^n})$$

- How small can $\mathbf{Adv}^{TPRP}_{\tilde{E}}(D,T)$ be? Cannot be better than $T/2^k$.
- Can have weaker security while designing TBC from BC.

TBC based on BC

Some Examples of TBC based on BC

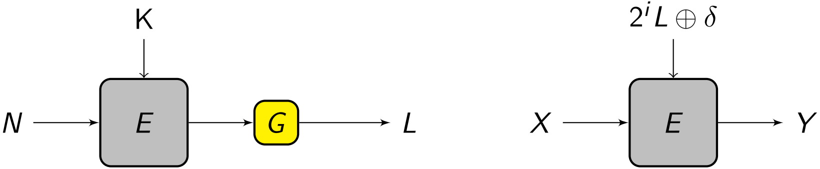


Figure: ICE1 with KDF1. (Remus-N1). Here tweak $= (N, i, \delta)$.

Some Examples of TBC based on BC

- D many queries to ICE1 with input 0^n and changing the tweak to get $Y_1, \ldots, Y_D$.
- $K_1, \ldots, K_D$: intermediate keys for the second call of BC.

Some Examples of TBC based on BC

- D many queries to ICE1 with input 0^n and changing the tweak to get $Y_1, \ldots, Y_D$.
- $K_1, \ldots, K_D$: intermediate keys for the second call of BC.
- Precompute T many blockcipher outputs $Y'_1, \ldots, Y'_T$ with input 0^n and key $K'_1, \ldots, K'_T$.

Some Examples of TBC based on BC

- D many queries to ICE1 with input 0^n and changing the tweak to get $Y_1, \ldots, Y_D$.
- $K_1, \ldots, K_D$: intermediate keys for the second call of BC.
- Precompute T many blockcipher outputs $Y'_1, \ldots, Y'_T$ with input 0^n and key $K'_1, \ldots, K'_T$.
- When $DT \approx 2^n$, we expect $K_i = K'_j$ (detectable through $Y_i = Y'_j$).

Some Examples of TBC based on BC

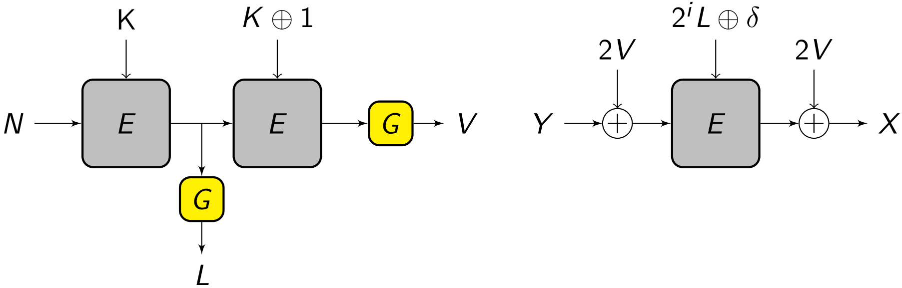


Figure: ICE2 with KDF2. (Remus-N2). Here tweak= (N, i, δ).

- TPRP advantage of ICE2 is $\frac{DT}{2^{2n}}$. Requires larger state.
- Can we have both (1) smaller state (2) higher security?

Figure: ICE1 with G as identity. (Remus-N1). Here tweak $= (N, i, \delta)$.

1. Use different reduction games considering μ-respecting adversary (the maximum number of query to TBC with same input is at most μ).
2. TPRP advantage of such an adversary against ICE1 is $\frac{\mu T}{2^n}$.
3. Restrict $\mu = O(n)$ and consider n-multicollision.

mixFeed

The mF Mode of AEAD

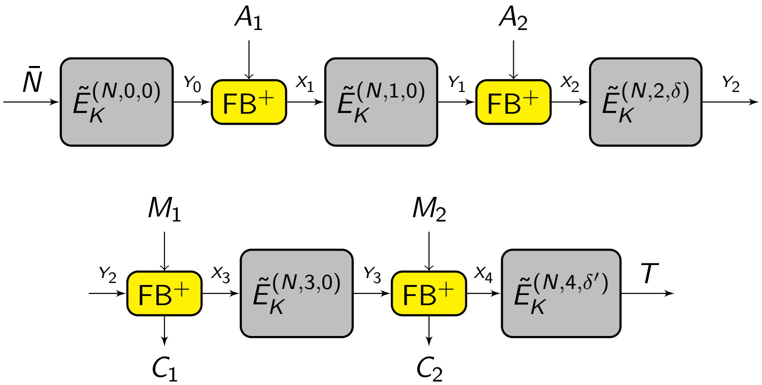
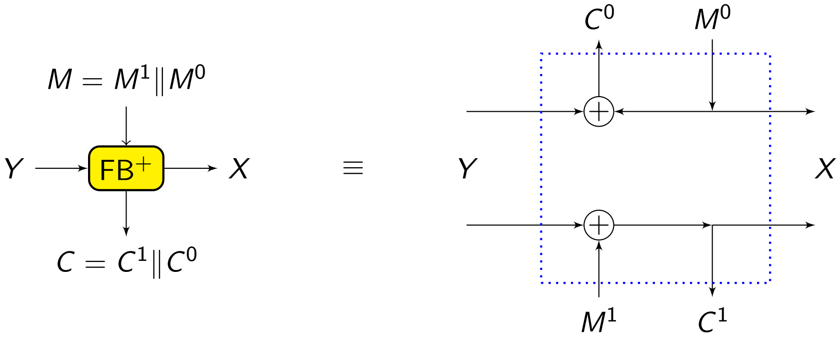


Figure: Block diagram of mF. $\mathrm{Fmt}_1(A) = (A_2 \| A_1), \mathrm{Fmt}_2(M) = (M_2 \| M_1)$.

FeedBack Function used in mF



The TBC in mixFeed

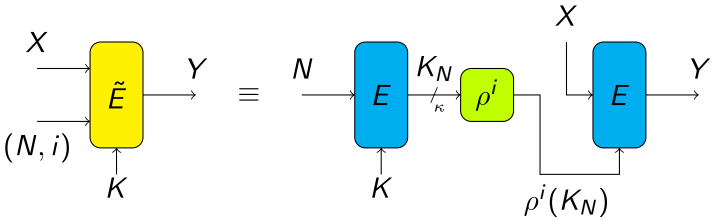


Figure: The tweakable block cipher in mixFeed. Here ρ is the 11-th round key function in AES key scheduling algorithm.

1. State size is just $n + k$ (i.e. 256).
2. Rate is 1.

- Domain Separation by Last block processing.

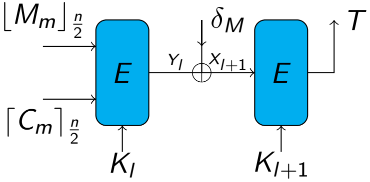


Figure: MixFeed Last block processing.

Security Definitions of Input Restricting TPRP

- Tweak space $\mathcal{T}$. n-bit TBC $\tilde{E}$. Tweakable random Permutation $\tilde{\Pi}$.
- μ-**TPRP**:
  - $\mathscr{A}^{\mathcal{O}}$, Restriction: $\forall X \in \{0,1\}^n$ number of queries $(\cdot, X) \leq \mu$
  - $\mathbf{Adv}_{\tilde{E}}^{\mu\text{-}TPRP}(\mathscr{A}) = \left| \Pr\left[\mathscr{A}^{\tilde{E}_K} = 1\right] - \Pr\left[\mathscr{A}^{\tilde{\Pi}} = 1\right] \right|$

$$\mathbf{Adv}_{\tilde{E}}^{\mu\text{-}TPRP}(q,t) = \max_{\mathscr{A}} \mathbf{Adv}_{\tilde{E}}^{\mu\text{-}TPRP}(\mathscr{A})$$

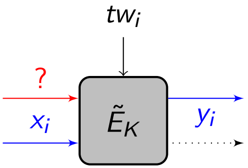
where max over all $\mathscr{A}$(number of queries $\leq q$, time $\leq t$).

(μ, D)-Multi-commitment-Prediction (or (μ, D)-mcp)

1. $\mathscr{A}^{\tilde{E}_K}$ runs in two phase. In the first phase it is μ-respecting.

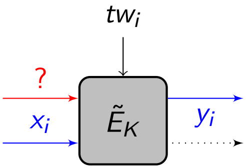
(μ, D)-Multi-commitment-Prediction (or (μ, D)-mcp)

1. $\mathscr{A}^{\tilde{E}_K}$ runs in two phase. In the first phase it is μ-respecting.
2. $\mathscr{A}$ commits D many (tw_i, x_i, y_i), $x_i, y_i \in \{0,1\}^{\frac{n}{2}}$.



(μ, D)-Multi-commitment-Prediction (or (μ, D)-mcp)

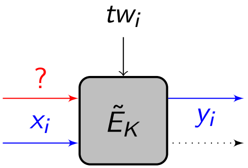
1. $\mathscr{A}^{\tilde{E}_K}$ runs in two phase. In the first phase it is μ-respecting.
2. $\mathscr{A}$ commits D many (tw_i, x_i, y_i), $x_i, y_i \in \{0,1\}^{\frac{n}{2}}$.



3. **Phase II:** $\mathscr{A}^{\tilde{E}_K}$ (with no restriction making at most D queries including prediction) predicts **fresh** some (tw_j, X_j, y_j) where $\lceil X_j \rceil_{\frac{n}{2}} = x_i$.

(μ, D)-Multi-commitment-Prediction (or (μ, D)-mcp)

1. $\mathscr{A}^{\tilde{E}_K}$ runs in two phase. In the first phase it is μ-respecting.
2. $\mathscr{A}$ commits D many (tw_i, x_i, y_i), $x_i, y_i \in \{0,1\}^{\frac{n}{2}}$.



3. **Phase II:** $\mathscr{A}^{\tilde{E}_K}$ (with no restriction making at most D queries including prediction) predicts **fresh** some (tw_j, X_j, y_j) where $\lceil X_j \rceil_{\frac{n}{2}} = x_i$.

4. $\mathscr{A}$ wins (μ, D)-mcp game if $\lfloor \tilde{E}_K(tw_j, X_j) \rfloor_{\frac{n}{2}} = y_j$, i.e. correctly predicts,

mcp Security Game

- $\mathbf{Adv}_{\tilde{E}}^{(\mu,D)\text{-}mcp}(\mathscr{A}) = \Pr\left[\mathscr{A} \text{ wins } (\mu, D)\text{-}mcp \text{ game }\right].$

$$\mathbf{Adv}_{\tilde{E}}^{(\mu,D)\text{-}mcp}(T) = \max_{\mathscr{A}} \mathbf{Adv}_{\tilde{E}}^{(\mu,D)\text{-}mcp}(\mathscr{A})$$

Where max over all $\mathscr{A}$ with runtime at most T (this includes the number of public primitive queries).

μ-Multicollision Game

- $\mathscr{A}^{\mathcal{O}_{\tilde{E}_K}}$
- $\mathscr{A}$ wins μ-multicollision game if
  - $\mathscr{A}$ makes μ many queries $(X_i, Y_i)_{i \in [1,\mu]}$ with $Y_i = Y_j \ \forall i, j \in [1, \mu]$ among all D queries.
- $\mathbf{Adv}_{\mathcal{O}}^{\mu\text{-}mult}(\mathscr{A}) = \Pr[\mathscr{A} \text{ wins } \mu\text{-multicollision game}]$

$$\mathbf{Adv}_{\mathcal{O}}^{\mu\text{-}mult}(D) = \max_{\mathscr{A}} \mathbf{Adv}_{\mathcal{O}}^{\mu\text{-}mult}(\mathscr{A})$$

Where max over all $\mathscr{A}$ (number of queries $\leq D$).

μ-Multicollision Game

Let P be the ideal n bit random permutation and P' is the $n/2$-bit truncated function of P.

Theorem

$$\mathbf{Adv}_{P'}^{\mu\text{-mcoll}}(D) \leq D\left(1 + \frac{\mu^2}{2^n}\right)\left(\frac{D}{2^{\frac{n}{2}}}\right)^{\mu-1}.$$

When $\mu = n$,

$$\mathbf{Adv}_{P'}^{n\text{-mcoll}}(D) = O\left(\frac{D}{2^{\frac{n}{2}}}\right).$$

Security Reductions of mixFeed

- $\mathscr{B}$: privacy adversary of mF.
  $\mathscr{A}$: μ-TPRP adversary of $\tilde{E}$.
  $\mathscr{C}$: multicollision adversary.

Theorem

$$\mathbf{Adv}_{mF}^{priv}(\mathscr{B}) \leq \mathbf{Adv}_{\tilde{E}}^{\mu\text{-}TPRP}(\mathscr{A}) + \mathbf{Adv}_{P}^{\mu+1\text{-mcoll}}(\mathscr{C}).$$

So,

$$\mathbf{Adv}_{\mathrm{mF}}^{priv}(D,T) \leq \mathbf{Adv}_{\tilde{E}}^{n\text{-}TPRP}(D,T) + O(D/2^{n/2}).$$

- For any (D, T) forging adversary $\mathscr{B}$ of mF we have.
- (i) $(\mu - 1, D)$-mcp adversary $\mathscr{A}$ and (ii) $\mathscr{C}$ with oracle $\mathcal{O}_{\tilde{E}_K}$ where $\mathcal{O}_{\tilde{E}}(tw, X, C) \rightarrow X' := C \oplus (0^{\frac{n}{2}} \| \lfloor \tilde{E}_K(tw, X) \rfloor_{\frac{n}{2}})$.

Theorem

For any forging adversary $\mathscr{B}$ of mFwith data complexity D there is (i) an $(\mu - 1, D)$-mcp adversary $\mathscr{A}$ of $\tilde{E}$, and (ii) an $\mu + 1$-multicollision adversary $\mathscr{C}$ as defined above, we have

$$\mathbf{Adv}_{mF}^{forge}(\mathscr{B}) \leq \mathbf{Adv}_{\tilde{E}}^{(\mu-1,D)\text{-}mcp}(\mathscr{A}) + \mathbf{Adv}_{\mathcal{O}_{\tilde{E}_K}}^{(\mu+1)\text{-mcoll}}(\mathscr{C}).$$

The TBC in mixFeed

(i) μ-respecting TPRP
(ii) (μ, D)-mcp advantage and
(iii) $(\mu + 1)$-multi-collision.

Assumption

For any $K \in \{0,1\}^n$ chosen uniformly at random, probability that K has a period at most l is at most $\frac{l}{2^{\frac{n}{2}}}$.

For random permutation the probability is much smaller: $\frac{l}{2^n}$.

The TBC in mixFeed

Theorem

Under the above assumption

$$\mathbf{Adv}_{\tilde{E}}^{(\mu,D)\text{-mcp}}(T) = O(\frac{D}{2^{\frac{n}{2}}}) + O(\frac{nT}{2^n})$$

$$\mathbf{Adv}_{\tilde{E}}^{n\text{-}TPRP}(D,T) = O\left(\frac{D}{2^{\frac{n}{2}}}\right) + O\left(\frac{nT}{2^n}\right).$$

$$\mathbf{Adv}_{\tilde{E}}^{n\text{-mcoll}}(D) \leq O(\frac{D}{2^{\frac{n}{2}}})$$

.

Security Bounds: mixFeed I

Theorem (Final Bound of mixFeed)

Under Assumption 1

$$\mathbf{Adv}^{priv}_{mixFeed}(D, T) = O\left(\frac{D}{2^{\frac{n}{2}}}\right) + O\left(\frac{nT}{2^n}\right)$$

$$\mathbf{Adv}^{forge}_{mixFeed}(D, T) = O\left(\frac{D}{2^{\frac{n}{2}}}\right) + O\left(\frac{nT}{2^n}\right)$$

Conclusion: mixFeed Mode of AEAD

- mixFeed is provable secure under the NIST requirements (by Assumption 1) in the nonce respecting scenario.
- As shown by Mustafa Khairallah (in the Forum), mixFeed is vulnerable to Nonce misuse attacks.
- the re-keying is done simply by the AES key scheduling algorithm and can be done online. So minimal state size.
- The Feedback function is extremely simple as it requires only n-bit XOR.

Conclusion: mixFeed Mode of AEAD

- mixFeed is provable secure under the NIST requirements (by Assumption 1) in the nonce respecting scenario.
- As shown by Mustafa Khairallah (in the Forum), mixFeed is vulnerable to Nonce misuse attacks.
- the re-keying is done simply by the AES key scheduling algorithm and can be done online. So minimal state size.
- The Feedback function is extremely simple as it requires only n-bit XOR.

Thank You!